

Europäisches Patentamt
European Patent Office
Office européen des brevets



(11) EP 0 772 127 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
07.05.1997 Bulletin 1997/19

(51) Int. Cl.⁶: G06F 11/20

(21) Application number: 96117200.4

(22) Date of filing: 25.10.1996

(84) Designated Contracting States:
DE FR GB

(30) Priority: 30.10.1995 JP 282072/95

(71) Applicant: HITACHI, LTD.
Chiyoda-ku, Tokyo 101 (JP)

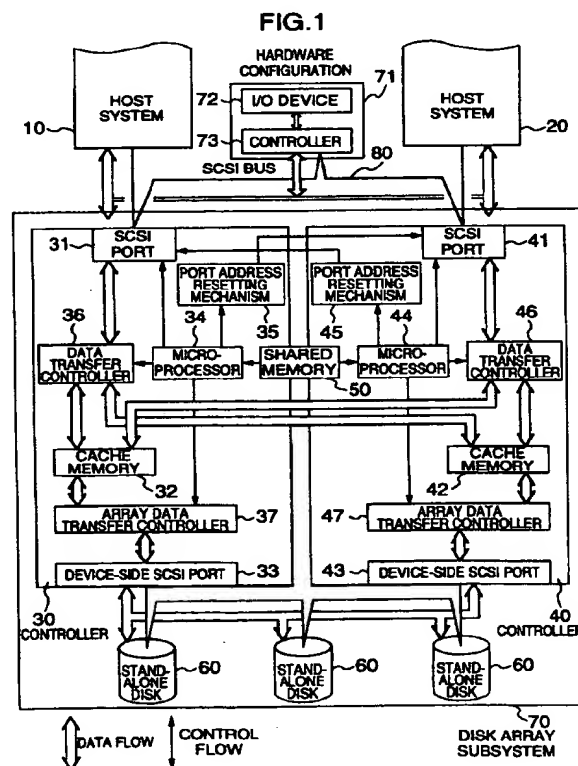
(72) Inventors:
• Murotani, Akira
Odawara-shi (JP)

• Nakano, Toshio
Chigasaki-shi (JP)
• Iwasaki, Hidehiko
Hiratsuka-shi (JP)
• Muraoka, Kenji
Odawara-shi (JP)

(74) Representative: Strehl Schübel-Hopf Groening &
Partner
Maximilianstrasse 54
80538 München (DE)

(54) Controller failure recovery in an external storage

(57) In an external storage, an I/O process is continued without any intervention of a user or a host system at failure of a controller. When a failure occurs in a controller (30), a host system 10 recognizes the failure of the controller (30). Before the failure is notified to the user and application to stop the job, the substitutive controller (40) reads the SCSI-ID possessed by an SCSI port (31) of the failed controller (30) from a shared memory (50), registers the SCSI-ID of the SCSI port (31) to the SCSI port (41) associated with the substitutive controller (40), and erases by a port address resetting facility 45 of the substitutive controller (40) the SCSI-ID possessed by an SCSI port (31) of the failed controller (30). Thanks to the provision, since the SCSI-ID specified at issuance of an I/O request is transferred between the controllers, the user or the host system need not alter the I/O request issuing route. Moreover, while the host system (10, 20) does not recognize the error, the transfer can be conducted.



EP 0 772 127 A1

Description

BACKGROUND OF THE INVENTION

The present invention relates to a technology to guarantee high reliability in operation of a plurality of controllers for input/output (I/O) devices in a computer system, and in particular, to a method of redundantly arranging controllers capable of transferring a process therebetween without intervention of the user and host systems at occurrence of a failure in one of the controllers in an external storage subsystem adopting a Small Computer Systems Interface (SCSI) in which the controllers are arranged at least in an duplicated configuration and the controllers can be accessed from the host systems.

In a system configuration employing the SCSI in which a plurality of controllers and a storage shared between at least two controllers are connected by an interface cable in a daisy chain to the host systems, the plural controllers respectively have different port addresses such as SCSI-IDs. Ordinarily, these controllers process I/O requests designated according to pertinent port addresses specified by the host systems.

Described in the JP-A-4-364514 is a system in which the controllers are arranged in the multiplex configuration such that I/O requests from a host apparatus to storages connected to the plural controllers are processed at a high speed. In such a conventional system, at occurrence of a failure in one of the controllers, when the host system alters the specification of the controller to execute the I/O request, it is possible that the I/O request is processed by a normal controller. However, in a system in which the host system and the plural controllers are connected to each other in a daisy chain, considerations have not been given to a procedure in which at occurrence of a failure occurs in a controller, the process is transferred to a normal controller for the execution thereof without intervention of the host system.

After issuing an I/O request to a controller, the host system ordinarily monitors termination of the I/O request by a timer in the host system. When the I/O is not terminated even when the monitor time predetermined by the host system lapses after the issuance of the I/O request, the host system assumes the state temporarily as an error. Conducting processes such as bus recovery process of an SCSI bus, the host system tries to issue again the same I/O request with specification of the port address of the failed controller.

When the controller does not respond to the re-issued I/O request, the host system regards the state as a permanent error and hence does not thereafter issue any I/O request to the failed controller. At failure of a controller in the conventional system, when the host system once recognizes the permanent error, the data process thereof is interrupted. Therefore, even there are disposed a plurality of controllers, the user intervention is required to continuously execute the data process of

the host system at failure of the pertinent controller.

Furthermore, in a case in which there are disposed a plurality of host systems, when a controller fails and enters a hang-up situation with the bus kept occupied by the failed controller, another data process being executed between another host system and another controller is also interrupted. The user intervention is also required to recover the interrupted data process.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a failure recovery method and system in which at occurrence of a failure in a controller, the process thereof is transferred to a normal controller to continuously achieve the data process without any intervention of the host system and user.

Additionally, in a case in which the failed controller has not yet received the I/O request from the host system and hence the error has not been assumed, it is necessary to possibly suppress I/O requests to the failed controller to prevent an abnormal operation. Consequently, in accordance with the present invention, the transfer of the port address and control information is executed after suppressing an event in which the host systems issue I/O requests thereto.

To achieve the object above according to the present invention, a normal controller has a function to receive control information of the failed controller and a function to reference the port address of the failed controller to add the contents thereof to the own port address. Furthermore, the normal controller possesses a function to reset the port address in the failed controller to thereby erase the port address.

Thanks to these functions, the normal controller can receive the port address and control information of the failed controller and accepts and executes the I/O request issued to the failed controller. In the operation, there may be employed a method in which the port address is reset by the pertinent failed controller.

Moreover, according to the present invention, there is disposed a function that the normal controller monitors a bus such as an SCSI bus at detection of the failure to thereby decide whether or not the failed controller has already received the I/O request from the host system. In a case in which the failed controller has already received the I/O request from the host system, the transfer of the port address and control information of the failed controller is terminated to prevent the host system from recognizing the permanent error so as to continue the process of the host system without any intervention of the user and host system.

In addition, when the normal controller is executing an I/O process at detection of a failure in a controller, it is assumed that the failed controller does not yet receive the I/O request from the host system. According to the present invention, there is provided a function to detect the condition such that the transfer of the port address and control information of the failed controller is accom-

plished during the I/O process execution of the normal controller.

As a result, I/O requests from the host system to the failed controller can be suppressed until the port address transfer process is completed. In addition, in a case in which the bus such as the SCSI bus is not being used by any controller at detection of the failure, it is considered the failed controller has not yet received the I/O request from the host system. According to the present invention, there is provided a function in which the condition is detected and the normal controller selects the failed controller such that the transfer of the port address and control information is executed after the selection is accomplished. Thanks to this function, I/O requests from the host system to the failed controller can be suppressed until the port address transfer process is completed.

Owing to adoption of the construction of this type, in a situation in which a failed controller have received an I/O request and the execution of the I/O process has not been terminated with a bus such as an SCSI bus kept exclusively reserved by the failed controller, a normal controller detects the state, completes reception of the port address and control information, and resets the failed controller within the I/O monitor time of the host system. This makes it possible that any subsequent I/O requests to the failed controller can be received for execution thereof by the normal controller. Resultantly, the system can respond to the I/O request re-issued from the host system and hence the interruption of the process of the host system as well as the inhibition of issuance of I/O requests from the host system can be prevented.

Moreover, at detection of a failure in a controller, the normal controller can suppress I/O requests from the host system to the failed controller. Therefore, in a case in which the failed controller has not yet received the I/O request, the host system need not recognize the error and any subsequent I/O requests can be received by the normal controller, thereby implementing the nonstop system operation.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects and advantages of the present invention will become apparent by reference to the following description and accompanying drawings wherein:

Fig. 1 is a hardware configuration diagram showing an embodiment of the present invention;

Fig. 2 is a diagram of processing sequence of host system at failure of a controller in the embodiment of Fig. 1;

Fig. 3 is a diagram briefly showing processed to be executed depending on states of the disk subsystem in the embodiment of Fig. 1;

Fig. 4 is a flowchart of processing executed at detection of the controller failure, specifically,

processing executed when the SCSI bus is in the bus free state in the embodiment of Fig. 1;

Fig. 5 is a flowchart of processing executed at detection of the controller failure, specifically, processing executed when the bus is in-use in the embodiment of Fig. 1;

Fig. 6 is a hardware configuration diagram of another embodiment according to the present invention; and

Fig. 7 is a schematic diagram showing a method of implementing the SCSI-ID transfer in the configuration of the embodiment of Fig. 6.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Description will now be given in detail of an embodiment according to the present invention.

In Fig. 1, reference numerals 10 and 20 indicate host systems as central processors to conduct data processing and a numeral 70 denotes a disk array subsystem as a peripheral unit in a dual controller structure. In the constitution of the disk array subsystem 70, a numeral 60 designates a standalone disk for storing therein data of the host systems, numerals 30 and 40 are controllers to supervise data transfers between the host systems and the standalone disks, a numeral 50 stands for a shared memory to transmit information between the controllers. A reference numeral 71 indicates another peripheral unit including an input/output (I/O) device 72 and a controller 73 to control the I/O device 72.

The host systems 10 and 20 are connected via an SCSI cable to the controllers 30, 40, and 73. In the constitution of the controller 30, a numeral 31 indicates an SCSI port to control an SCSI bus on the host system side, a numeral 32 is a cache memory, a numeral 33 denotes a device-side SCSI port to control the SCSI bus connecting the standalone disks to the controller 30, a numeral 34 designates a microprocessor to control overall operations of the controller 30, a numeral 35 is a port address resetting facility to reset the SCSI port of the controller 40, a numeral 36 is a data transfer controller to execute a data transfer between the host system 10 and the cache memory 32, and a numeral 37 indicates an array data transfer controller to execute a data transfer between the cache memory 32 and the standalone disk 60.

The data transfer controller 36 has a function to write, when transferring data to the cache memory 32, the contents of data also in the cache memory 42 of the controller 40. In addition, the array data transfer controller 37 possesses a function to generate redundant data for data buffered in the cache memory 32. This function can also be employed to restore data.

The controllers 40 and 30 mutually have the same configuration. Specifically, for each a constituent element of the controller 30, a reference number obtained by adding ten to the reference number of the constituent

element indicates a partner or associated constituent element in the controller 40. The port address resetting facility 45 can reset the SCSI port 31 of the controller 30. The port address resetting facilities 35 and 45 reset port addresses, i.e., SCSI-IDs preserved by the SCSI ports 41 and 31 in the respective controllers 40 and 30. According to the SCSI standards, the SCSI-IDs can be erased in the next arbitration phase.

In addition, since the data transfer controller 36 has a function to write data in the cache memory 32, any data items transferred from the host systems 10 and 20 are duplicatedly buffered in the respective cache memories 32 and 42. Thanks to the provision, even when a failure occurs in one of the controllers, the remaining controller can receive the process of the failed controller to execute the process using its own data in the cache memory.

The I/O process flow will be described according to an example in which the host system 10 achieves a data transfer via the controller 30. The system 10 issues an I/O request with an SCSI-ID designating the controller 30. In the controller 30, the SCSI port 31 keeping therein the SCSI-ID receives the I/O request and then passes the request to the microprocessor 34. The microprocessor 34 analyzes the I/O request and then instructs the data transfer controller 36 to execute a data transfer between the system 10 and the disk 60.

The transfer data is provisionally buffered in the cache memory 32 and is then written also in the cache memory 42 for a possible failure in the controller 30. In this connection, the SCSI-ID is set by the microprocessor 34 at initialization of the SCSI port 31, for example, when the system is powered. The SCSI-ID is saved in the shared memory 50 at the same time. Stored in the shared memory 50 is also control information so that the process can be continuously executed by a normal controller when one of the controller system fails in the dual controller configuration.

Referring now to the process sequence of the host system at failure of the controller shown in Fig. 2, description will be given of a method of continuing an I/O operation of the host system according to the present invention.

First, the internal construction of the host system will be described. In Fig. 2, a numeral 81 is an application program for executing data processing to achieve various requests from the user, a numeral 82 denotes a file system for keeping therein data structure and controlling I/O requests, a numeral 83 indicates a device driver for converting an I/O request into a request mode suitable for a peripheral unit, a numeral 84 stands for an SCSI card for transmitting an I/O request to the SCSI bus, a numeral 85 is a transfer I/O buffer, and a numeral 86 designates a system log in which failure information of the host systems is accumulated.

Next, description will be generally given of the processing of the host system 10 when a failure occurs in the controller 30 of the disk subsystem. Receiving an I/O request occurring in the application 81, the file sys-

tem 82 issues an I/O request to the SCSI bus 80 via the device driver 83 and SCSI card 84. On receiving the request, when the controller 30 detects a failure in the disk subsystem, the controller 30 reports Check Condition for the I/O request.

Next, the device driver 83 issues a Request Sense command to receive Sense Data which is detailed failure information. According to the Sense Data, the device driver 83 recognizes the state of the controller 30. As a result, the driver 83 issues again (retries) the same I/O request. Since the failed controller 30 cannot either execute the re-issued I/O request, the device driver 83 instructs an operation to discard the process associated with the I/O request and repeats the operation, for example, by Retry after an Abort message. After this operation, the driver 83 recognizes the state as a permanent error to notify the condition to the file system 82.

Receiving the permanent error report, the file system 82 does not thereafter issue any I/O request to the disk subsystem 70. The file system 82 then erases non-reflection data of the I/O buffer 85 and records a failure occurrence in the system log, and then sends an error message via the application 81 to the user. Consequently, the integrity of updated data cannot be preserved between the application 81, file system 2, and disk subsystem depending on cases. In consequence, in any case to which the present invention is not applied, the user is required to once stop the application and the like to restore the disk subsystem so as to thereafter execute again a sequence of processes possibly having caused the mismatching of data in the host system.

As another example of general processing, there exists a case in which the controller 30 can not report Check Condition to the device driver 83 even at failure. Namely, the controller 30 does not notify the occurrence of the failure to the device driver 83. On this occasion, the device driver 83 checks the state of the disk subsystem by monitoring the state according to a fixed period of time indicated by a timer. When the response is not received within the fixed period of time, the device driver conducts, like in the example above, the process beginning at the re-issuance (retry) of the same I/O request.

Referring to Fig. 1, description will be given of an advantageous feature in which the I/O process can be continued without conducting the user operation in accordance with the present invention. The controllers 30 and 40 update monitor information items of the respective controllers in the shared memory 50 at a fixed interval of time; moreover, mutually reference monitor information thereof.

In a case in which the controllers 30 and 40 are respectively receiving I/O requests issued respectively from the host systems 10 and 20, when a failure occurs in the controller 30, the monitor information of the controller 30 in the shared memory 50 is updated by the controller 30 to information indicating the failure. Or, the information is not updated even when a fixed period of time lapses. Referencing the monitor information in the

shared memory 50, the controller 40 detects the failure of the controller 30, reads the SCSI-ID of the SCSI port 31 and control information of the controller 30 from the shared memory 50, and adds by the microprocessor 44 the SCSI-ID of the SCSI port 31 to the SCSI port 41.

Additionally, using the SCSI port resetting facility 45, the controller 40 erases the SCSI-ID possessed by the SCSI port 31. This enables the SCSI port 41 to accept an I/O request issued from the host system 20 and one issued from the host system 10 so that the retry of the host system 10 is received for execution thereof by the controller 40.

When the retry is normally executed, a normal execution of the I/O request is reported to the file system 82 and the processing of the host system 10 is normally continued. The control information includes transit information in relation to transfers of data from the cache memories 32 and 42 to standalone disks. Consequently, receiving the control information, the controller 40 can transfer, in place of the controller 30, the duplicated data written in the cache memory 42 as alternative data of the Write data kept remained as non-reflection data in the cache memory 32.

Since the method of failure detection and control information transfer of the controller 30 is not the inherent characteristic of the present invention and has already been described in detail in the Japanese Patent Application No. 7-139781 (filed on June 7, 1995) by the applicant of the present invention and hence description thereof will be avoided.

For the transfer by the controller 40 of the SCSI-ID of the SCSI port 31 to the SCSI port 41 and the transfer of control information of the controller 30 to the controller 40 described above, the associated processing is required to be appropriately accomplished according to the state of the controller 30. Otherwise, the transfers cannot be correctly carried out. According to the present invention, the status of the failed controller 30, more specifically, the state of reception by the failed controller 30 of the I/O request from the host system is decided on the basis of the usage state (signal state) of the SCSI bus.

In the following examples, description will be given of a case in which a failure takes place in the controller 30 of Fig. 1 and the process is continued by the normal controller 40.

Referring next to Fig. 3, description will be given of a process to be executed according to the state of the disk subsystem.

In general, it is difficult to completely forecast operation to be achieved by the failed controller when an I/O request is received from the host system. Therefore, in a case in which the failed controller 30 did not yet receive the I/O request from the host system 10 when the failure of the controller 30 is detected by the controller 40, the transfer process of the SCSI-ID including the addition of the SCSI-ID to the SCSI port 41 and the resetting of the SCSI port 31 is executed as early as possible so that the controller 40 receives the I/O

request.

However, when an I/O request is issued from the host system 10 with specification of the SCSI-ID during the transfer process of the SCSI-ID, the controllers 30 and 40 possess the same SCSI-ID and hence the operation of the SCSI bus becomes unstable. In this situation, according to the present invention, there is provided a method in which the SCSI bus 80 is dedicatedly occupied by one controller during the SCSI-ID transfer process so as to suppress the I/O request issuance from the host system 10.

In accordance with the present invention, the controller 40 monitors the utilization status (signal state) of the SCSI bus 80 to decide whether or not the controller 30 has already received the I/O request from the host system 10, thereby executing a process associated with the decision.

In one of the utilization statuses of the SCSI bus 80, the SCSI bus 80 is possibly in the bus free state when a failure is detected in the controller 30. In this case, the SCSI bus 80 is possibly in the bus free state. Since the controller 30 has not received yet the I/O request, the controller 40 executes a host operation (the initiator operation) such that the controller 40 selects the controller 30 to exclusively occupy the SCSI bus 80. This makes it possible to suppress the issuance of an I/O request from the host system 10 such that the controller 40 conduct the transfer of the SCSI-ID during this period.

In one of the utilization statuses of the SCSI bus 80, it may be possible that the controller 40 is executing an I/O process through the SCSI bus 80 when a failure is detected in the controller 30. In this situation, it may be possible that the controller 40 is executing an I/O process through the SCSI bus 80. On this occasion, the controller 30 has not received the I/O request and hence the SCSI bus 80 is set to the bus free state at termination of the I/O process and an I/O request may possibly be issued from the host system 10. To overcome this difficulty, the controller 40 completely executes also the SCSI-ID transfer during the execution of the pertinent I/O process. If the SCSI-ID transfer is not completed during the execution of the pertinent I/O, the controller 40 does not send the report of the I/O termination status until the ID transfer is completely finished.

In one of the utilization statuses of the SCSI bus 80, the SCSI bus is possible being used when a failure is detected in the controller 30. In this case, the system is in a state in which the arbitration or selection is being executed according to the SCSI standards, a state in which another SCSI device connected to the SCSI bus 80 is using the SCSI bus 80, or a state in which the controller 30 has already received the I/O request from the host system 10.

In this situation, the controller 40 monitors the BSY signal of the SCSI bus 80. In association with the monitor period, when the BSY signal continues for a period of time equal to or more than the period of time in which the arbitration phase is changed via the selection phase

to the message out phase according to the SCSI standards, it can be decided that the signal is the BSY signal indicating an I/O process in execution, not the BSY signal of the bus mastership arbitration. After the signal decision, the controller 40 executes the SCSI-ID transfer process at a high speed.

If another SCSI device is using the SCSI bus 80, the controller 30 has not received the I/O request. Therefore, the controller 40 achieves the transfer process at a high speed while another SCSI device is using the SCSI bus 80.

If the controller 30 has already received the I/O request from the host system 10, the failed controller 30 has already stopped its operation with the SCSI bus 80 exclusively possessed by the controller 30. Since the device driver 83 is monitoring the I/O operation by the internal timer, the controller 40 is required to execute the SCSI-ID transfer before the host system 10 conducts the Bus Reset and Retry so that the controller 40 responds to the Retry. The monitor period of the controller 40 to monitor the SCSI bus 80 is fully shorter than the I/O process monitor period of the host system 10. Consequently, the controller 40 is required to completely achieve the SCSI-ID transfer prior to the bus resetting indication from the host system. This can be satisfactorily achieved thanks to the provision above.

Referring to Figs. 4 and 5, description will be given of a procedure to acquire the state of the disk subsystem by monitoring the SCSI bus and an associated procedure of transferring the SCSI-ID.

Description will be given of a case in which the SCSI bus 80 is in the bus free state when a failure of the controller 30 is detected by the controller 40 in Fig. 4.

Since the SCSI bus 80 is in the bus free state (step 400), the controller 40 recognizes that the controller 30 has not yet received the I/O request from the host system 10. The controller 40 then instructs the SCSI port 41 to start the initiator operation to participate in the arbitration of the SCSI bus 80 (step 401).

As a result, when the controller 40 remains in the arbitration (yes in step 402), the controller 40 specifies in the selection phase the SCSI-ID of the SCSI port 31 of the failed controller 30. In this situation, even if a failure occurs in the controller 30, the SCSI port 31 normally functions in most cases. Consequently, there is set a state in which the SCSI port 31 of the controller 30 exclusively occupies the SCSI bus 80 (step 404). In this state, the controller 40 adds the SCSI-ID possessed by the SCSI port 31 to the SCSI port 41 (step 405) and then resets the SCSI port 31 (step 406). The SCSI bus 81 exclusively occupied by the controller 30 is released by resetting the SCSI port 31 and is returned to the bus free state. Thereafter, the controller 40 receives the I/O request from the host system 10 (step 413). The I/O process can be continued in this way without any intervention of the user.

When the controller 40 cannot remain in the arbitration (no in step 402), it is decided whether or not the controller 40 is selected by the host system 20 in the

selection phase (step 403). If the controller 40 is selected by the host system 20 (yes in step 403), there is set a state in which the controller 40 exclusively occupies the SCSI bus 80. In this state, the controller 40 receives the I/O request from the host system 20 (step 407) and then provisionally interrupts the processing. The controller 40 adds the SCSI-ID possessed by the SCSI port 31 to the SCSI port 41 (step 408) and then resets the SCSI port 31 (step 409). After resetting the port 31, the controller 40 executes the I/O request from the host system 20 (step 410) and then restores the SCSI bus 80 to the bus free state. After this point, the controller 40 receives the I/O request from the host system 10 (step 413).

If the controller does not remain in the arbitration (no in step 402) and is not selected by the host system 20 (no in step 403), the controller 40 assumes a state in which the controller 30 having received the I/O request from the host system 10 or another SCSI device exclusively occupies the SCSI bus 80. In this situation, while the state is kept unchanged, the controller 40 adds the SCSI-ID possessed by the SCSI port 31 (step 411) to the SCSI port 41 and then resets the SCSI port 31 (step 412). If the controller 30 exclusively occupies the SCSI bus 80, the SCSI bus 80 is restored to the bus free state by resetting the SCSI port 31. If another SCSI device exclusively occupies the SCSI bus 80, the SCSI bus 80 is restored to the bus free state when the I/O process of the SCSI device is terminated. Thereafter, the controller 40 accepts the I/O request from the host system 10 (step 413).

Referring next to Fig. 5, description will be given of a processing procedure in a case in which the BSY signal of the SCSI bus 80 is asserted at detection of the failure of the controller 30 (step 500).

The controller 40 first determines whether or not the controller 40 is executing an I/O request from the host system 20 (step 501). If this is not the case (no in step 501), the controller 40 continuously monitors the state of the SCSI bus 80 for a period of time equivalent to the period in which the arbitration phase according to the SCSI standards is changed via the selection phase to the message out phase (step 502).

At detection of the failure, if the controller 40 is executing an I/O operation (yes in step 501) or the controller 40 is selected by the host system 20 during the monitor operation of the SCSI bus 80 (left branch in step 502), there is assumed a state in which the SCSI bus 80 is exclusively occupied by the controller 40 and the controller 30 has not received the I/O request. In this state, prior to reporting the termination status of the I/O execution (step 503), the controller 40 adds the SCSI-ID possessed by the SCSI port 31 to the SCSI port 41 (step 504) and then resets the SCSI port 31 (step 505). After resetting the port 31, the controller 40 notifies the I/O termination status and then terminates the I/O operation (step 506).

The SCSI bus 80 is set to the bus free state when the I/O execution process is terminated, and the control-

ler 40 receives any subsequent I/O request from the host system 10. In this fashion, it is possible to continuously execute the I/O process without user intervention.

When the bus free state is detected during the monitor operation of the SCSI bus 80 (central branch in step 502), the process at bus free detection of Fig. 4 is executed.

If the controller 40 is not executing an I/O operation and the SCSI bus 80 is not released during the monitor operation (right branch in step 502), the controller 40 recognizes that the controller 30 or another SCSI device exclusively occupying the SCSI bus is executing an I/O operation. Continuing the SCSI bus monitoring operation (step 508), the controller 40 adds the SCSI-ID possessed by the SCSI port 31 to the SCSI port 41 (step 509) and then resets the SCSI port 31 (step 510).

When the controller 30 exclusively occupies the SCSI bus 80, the bus 80 is returned to the bus free state by resetting the SCSI port 31. When another SCSI device exclusively occupies the SCSI bus 80, the bus 80 is returned to the bus free state when the I/O operation of the SCSI device is terminated. Thereafter, the controller 40 receives the I/O request from the host system 10. If the bus is released before the SCSI port 31 is completely reset (broken line in step 508), there is executed the process at detection of the bus free state shown in Fig. 4.

As a result of the processing procedure, the I/O request from the host system 10 can be executed by the controller 40 when a failure occurs in the controller 30, thereby preventing the permanent error. Consequently, the data processing of the system 10 can be normally continued.

Referring next to Figs. 6 and 7, description will be given that the present invention can be implemented in a configuration of the controller not including the port address resetting facility.

Fig. 6 is a diagram showing the configuration developed by removing the port address resetting facility from the controller of Fig. 1. Numerals 90 and 100 indicate controllers respectively conducting functions of the controllers 30 and 40 of Fig. 1 and a numeral 50 indicates a shared memory to supply information between the controller 90 and 100.

In an internal constitution of the controller 90, a numeral 34 is a microprocessor controlling overall operation of the controllers, a numeral 31 indicates an SCSI port which can be controlled only by the microprocessor 34, a numeral 32 denotes a cache memory, a numeral 33 stands for a device-side SCSI port, a numeral 36 designates a data transfer controller, and a numeral 37 is an array data transfer controller. The controllers 100 and 90 are of the same configuration. In the following paragraphs, description will be given of an example in which the controller 90 receives an I/O request from the host system 10 of Fig. 1 and the controller 100 receives an I/O request from the host system of Fig. 1. Fig. 7 is a diagram showing an SCSI-ID transfer processing procedure with its abscissa representing lapse of time.

When a failure occurs in the controller 90, the controller 100 detects the failure and then sets at a particular address in the shared memory 50 a failure flag indicating the occurrence of the failure in the controller 90. Thereafter, the controller 100 reads the SCSI-ID of the SCSI port 31 and control information of the controller 90 from the shared memory 50, and adds by the microprocessor 44 the SCSI-ID to the SCSI port 41. In contrast thereto, the controller 90 recognizes its own failure according to the failure flag in the shared memory 50 and enters a wait state in which by use of an internal timer, the controller does not execute its operation for a period of time equivalent to the period of time in which the transfer processing of the controller 100 is completely executed.

The controller 90 determines through the wait operation the completion of the processing of the controller 100 and then erases by the microprocessor 34 the SCSI-ID possessed by the SCSI port 31. As a result, the SCSI-ID transfer process is terminated and then the SCSI port 41 is enabled to receive the I/O request from the host system 10 of Fig. 1.

Since the SCSI-ID process can be conducted without using the port address resetting facility as above, the present invention is effective also in the configuration not including the port address resetting facility. It is to be assumed that also in a case in which a failure occurs in the controller 90, the microprocessor 34 and SCSI port 31 function normally.

While the present invention has been described with reference to the particular illustrative embodiments, it is not to be restricted by those embodiments but only by the appended claims. It is to be appreciated that those skilled in the art can change or modify the embodiments without departing from the scope and spirit of the present invention.

Claims

1. A failure recovery method for use in a data processing system including at least one host system (10, 20), a plurality of controllers (30, 40, 73, 90, 100), and an interface cable (80) connecting said host system to said controllers in a daisy chain, said controllers respectively including therein I/O ports (31, 41) being connected to said interface cable and having mutually different IDs (SCSI-IDs), an I/O device being controlled by a group of at least two controllers (30, 40, 90, 100), the method comprising the steps of:

detecting, when a failure is detected in a controller (30, 90) of said group, a utilization state of said interface cable by a controller (40, 100) as a substitutive unit of a failed controller (39, 90) of said group;
deciding, according to the utilization state of said interface cable, a state of reception by said failed controller of an I/O request from said host

system;

suppressing by a substitutive controller, when the I/O request is not yet received by said failed controller as a result of the decision, reception of the I/O request by said failed controller; adding an ID of an I/O port (31) related to said failed controller to an I/O port (41) of said substitutive controller; and resetting the I/O port related to said failed controller; and adding by said substitutive controller, when the I/O request is already received by said failed controller as a result of the decision, the ID of said I/O port related to said failed controller to the I/O port of said substitutive controller and resetting the I/O port related to said failed controller before said host system recognizes a permanent error in said failed controller.

2. A failure recovery method according to Claim 1, wherein, in resetting the I/O port related to said failed controller, reset is carried out by hardware resetting means (45) in said substitutive controller.
3. A failure recovery method according to Claim 1, wherein, in resetting the I/O port related to said failed controller, said substitutive controller further includes the steps of:

indicating to said failed controller to reset the I/O port related to said failed controller after lapse of a predetermined period of time; and adding the ID of the I/O port related to said failed controller to the I/O port of said substitutive controller within said predetermined period of time.

4. A failure recovery method according to Claim 1, wherein said interface cable is a Small Computer Systems Interface (SCSI) bus cable.
5. A data processing system, comprising:

at least one host system (10, 20);
a plurality of controllers (30, 40, 73; 90, 100);
and
an interface cable (80) connecting said host system to said controllers in a daisy chain, said controllers respectively including therein I/O ports (31, 41) being connected to said interface cable and having mutually different IDs (SCSI-IDs);
an I/O device being commonly controlled by a group of at least two controllers (30, 40; 90, 100); and
a shared memory (50) being commonly accessed from said group, each of controllers in said group including a microprocessor, the microprocessor in each of said controllers including:

means (44) for detecting a failure in a controller (30, 90) of said group according to contents of said shared memory;

means (44) for detecting a utilization state of said interface cable via an I/O port;
means (44) for deciding, according to the utilization state of said interface cable, a state of reception by said failed controller of an I/O request from said host system;

means (44) for suppressing, when the I/O request is not yet received by said failed controller as a result of the decision, reception of the I/O request by said failed controller; adding an ID of the I/O port (31) related to said failed controller to an I/O port (41) of a controller of its own; and indicating to reset the I/O port related to said failed controller; and

means for adding, when the I/O request is already received by said failed controller as a result of the decision, the ID of the I/O port related to said failed controller to the I/O port of the controller of its own; and indicating to reset the I/O port related to said failed controller before said host system recognizes a permanent error in said failed controller.

6. A data processing system according to Claim 5, wherein each of the controllers of said group includes hardware resetting means (45) responsive to an indication from said reset indicating means for resetting the I/O port related to said failed controller.

7. A data processing system according to Claim 5, wherein:

said reset indicating means writes a failure flag at a predetermined address in said shared memory, said flag indicating an occurrence of a failure;

a processor in said failed controller functions as means for reading said failure flag from said shared memory and resetting the I/O port related thereto after lapse of a predetermined period of time; and

said reset indicating means adds the ID of the I/O port related to said failed controller to the I/O port related to own controller within said predetermined period of time.

8. A data processing system according to Claim 5, wherein said interface cable is an SCSI bus cable.
9. An external storage for use in a data processing system including a host system (10, 20), an external storage (70) including a plurality of controllers (30, 40, 73; 90, 100) respectively having therein ports possessing identifiers (IDs) as individual port addresses and a group of storages (60) controlled

by and shared between said plural controllers, and an interface cable (80) connecting in a daisy chain said host system to said plural controllers having the ports therein, said plural controllers and storages being accessible from said host system,

said external storage having a function that at occurrence of a failure in a controller excepting at least one controller, a normal controller detects the failure, references a port address of a failed controller, receives control information of said failed controller, and adds control information to the port address thereof.

10. An external storage according to claim 9, further including a shared memory (50) for each of said plural controllers for storing therein the port address and control information of each of said controllers and thereby transmitting information between said controllers.

11. An external storage in a data processing system including host system (10, 20), an external storage (70) including a plurality of controllers (30, 40, 73; 90, 100) respectively having therein ports possessing identifiers (IDs) as individual port addresses and a group of storages (60) controlled by and shared between said plural controllers, and an interface cable (80) connecting in a daisy chain said host system to said plural controllers having the ports therein, said plural controllers and storages being accessible from said host system,

said external storage having a function that at occurrence of a failure in a controller excepting at least one controller, a normal controller detects the failure, references a port address of a failed controller, receives control information of said failed controller, and adds the control information to the port address thereof,

a controller having a port address resetting facility (45) for resetting the port address of said failed controller and erasing an ID thereof in such a manner that the controller resets the port address of said failed controller, that said failed controller does not respond to subsequent I/O requests from said host system, and that said normal controller having received the port address responds to the I/O requests.

12. An external storage according to claim 11, wherein, at occurrence of the failure in the controller, in a state in which said host system has not executed an I/O request to said failed controller and said interface cable connecting said host system to said controllers is not being used,

a normal controller executes selection for said failed controller to acquire a bus mastership between said normal controller and said failed controller, thereby suppressing issuance of an I/O request from said host system to said failed control-

ler during a transfer process of the port address by said normal controller.

13. An external storage according to claim 11, wherein, at occurrence of the failure in the controller, in a state in which said host system has not executed an I/O request to said failed controller and said normal controller is using the bus, said normal controller completes the transfer process of the port address of said failed controller during the processing of the I/O request issued from said host system and then notifies termination of the I/O request, thereby suppressing issuance of an I/O request from said host system to said failed controller during the transfer process of the port address by said normal controller.

14. An external storage according to claim 11, wherein;

said interface cable is an SCSI cable;

said normal controller monitors, when the bus is in use at occurrence of the failure in the controller, a BSY signal of the bus to determine whether or not the bus is being used by another device connected to the bus, whether or not the system is in a transit state from an arbitration phase to a selection phase according to the SCSI standards, and whether or not said failed controller already received an I/O request from said host system,

said normal controller executes, when the bus is released during the monitor operation, selection for said failed controller to attain a bus mastership between said normal and failed controllers,

said normal controller completes, when said normal controller is selected during the monitor operation, the transfer process of the port address of said failed controller during the processing of the I/O request issued from said host system and then notifies termination of the I/O request, and

said normal controller terminates during the monitoring period the transfer process of the port address of said failed controller.

15. An external storage according to claim 14, wherein the monitoring period of the bus mastership is set to be equal to or more than a period of time in which the arbitration phase is changed via the selection phase to a message out phase according to the SCSI standards so as to confirm that the BSY signal is not associated with arbitration of the bus mastership but is caused by an I/O execution process, thereby executing the transfer of the port address of said failed controller.

16. An external storage in a data processing system including a host system (10, 20), an external stor-

age (70) including a plurality of controllers (30, 40, 73; 90, 100) respectively having therein ports possessing identifiers (IDs) as individual port addresses and a group of storages (60) controlled by and shared between said plural controllers, and an interface cable (80) connecting in a daisy chain said host system to said plural controllers having the ports therein, said plural controllers and storages being accessible from said host system, wherein:

at occurrence of a failure in a controller excepting at least one controller, a failed controller recognizes the failure thereof and enters a wait state without executing a control operation thereof in at least a period of time equal to time in which said normal controller conducts a transfer process of control information of said failed controller and addition of a port address; after said normal controller which recognized the failure finishes the transfer and addition processes, said failed controller erases the port address of said failed controller; and said normal controller which received the port address of said failed controller responds to a subsequent I/O request issued from said host system since the port address of said failed controller is already erased.

17. An external storage according to claim 16, wherein at occurrence of the failure in the controller, in a state in which said host system has not executed an I/O request to said failed controller and said interface cable connecting said host systems to said controllers is not being used, said normal controller executes selection for said failed controller to acquire a bus mastership between said normal controller and said failed controller, thereby suppressing issuance of an I/O request from said host system to said failed controller during the transfer process of the port address by said normal controller.

18. An external storage according to claim 16, wherein, at occurrence of the failure in a controller, in a state in which a host system has not executed an I/O request to said failed controller and said normal controller is using the bus, said normal controller completes the transfer process of the port address of said failed controller during the processing of the I/O request issued from said host system and then notifies termination of the I/O request, thereby suppressing issuance of an I/O request from said host system to said failed controller during the transfer process of the port address by said normal controller.

19. An external storage according to claim 16, wherein:

when the bus is in use at occurrence of the failure in the controller, said normal controller monitors a BSY signal of the bus to determine whether or not the bus is being used by another device connected to the bus, whether or not the system is in a transit state from an arbitration phase to a selection phase according to the SCSI standards, and whether or not said failed controller already received the I/O request from said host system;

when the bus is released during the monitor operation, the normal controller executes selection for said failed controller to attain a bus mastership between said normal and failed controllers;

when said normal controller is selected during the monitor operation, said normal controller completes the transfer process of the port address of said failed controller during the processing of the I/O request issued from said host system and then notifies the termination of the I/O request; and

said normal controller terminates during the monitoring period the transfer process of the port address of said failed controller.

20. An external storage according to claim 16, wherein the monitoring period of the bus mastership is set to be equal to or more than a period of time in which the arbitration phase changes via the selection phase to a message out phase so as to confirm that the BSY signal is not associated with arbitration of the bus mastership but is caused by an I/O execution process, thereby executing the transfer of the port address of said failed controller.

21. A host system and an external storage connected by an interface cable in a configuration including a host system, an external storage including a plurality of controllers respectively having therein ports possessing identifiers (IDs) as individual port addresses and a group of storages controlled by and shared between said plural controllers, and an interface cable connecting in a daisy chain said host system to said plural controllers having the ports therein, said plural controllers and said storages being accessible from said host system,

said external storage having a function that at occurrence of a failure in a controller excepting at least one controller, said normal controller detects the failure, references the port address of the failed controller, receives control information of said failed controller, and adds the control information to the port address thereof, said host system having a function that in a state in which a controller having received an I/O request issued from the host system cannot respond thereto due to occurrence of a failure

in the controller, said host system monitors an I/O completion report from the controller, issues again the I/O request to said failed controller after lapse of the predetermined monitoring period, executes a recovery process including a resetting operation, recognizes a permanent error when the controller does not respond to the recovery process, and notifies the error to the application, and

said normal controller completing an operation including the reference, transfer, and additional port address processes before the permanent error is recognized, thereby preventing a report of the permanent error to an application of said host system.

5

10

15

20

25

30

35

40

45

50

55

FIG. 1

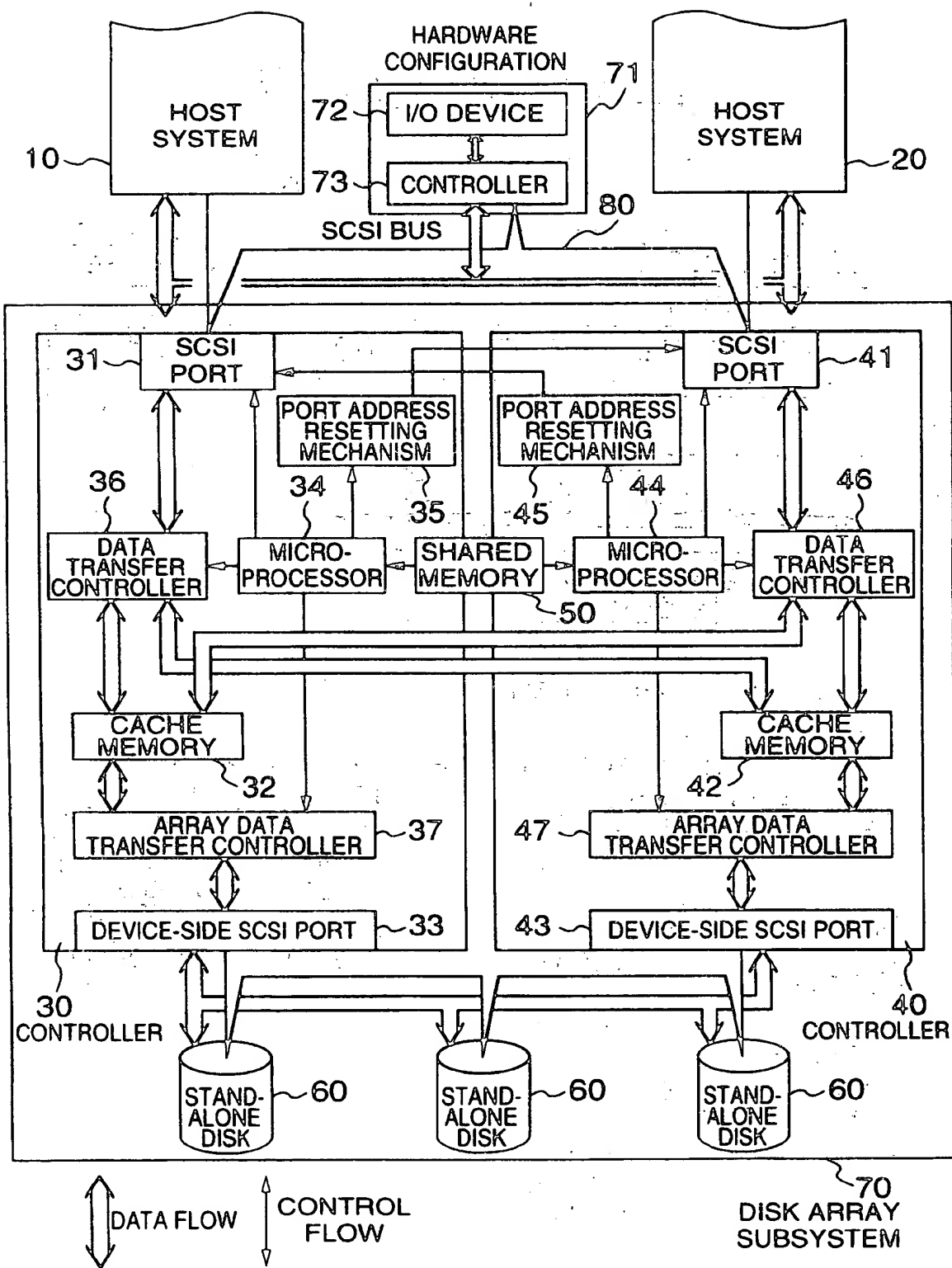
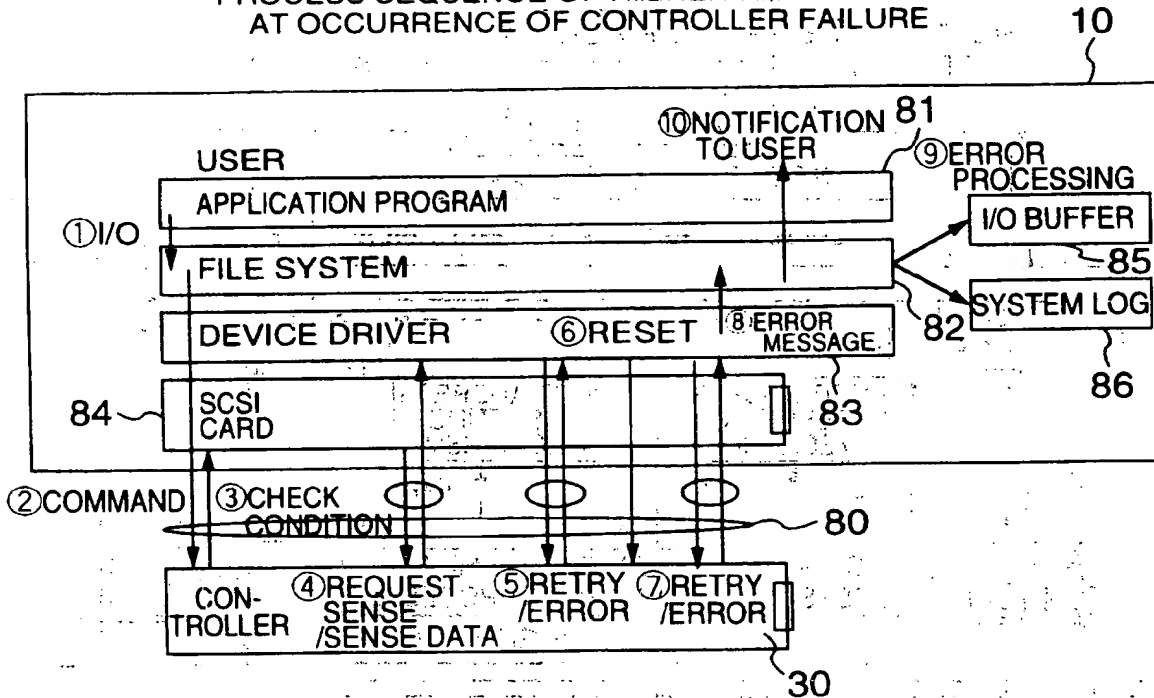


FIG.2

PROCESS SEQUENCE OF HIGHER-RANK SYSTEM
AT OCCURRENCE OF CONTROLLER FAILURE



① APPLICATION ISSUES I/O REQUEST

② ISSUE COMMAND

③ CHECK CONDITION (ERROR)

④ ISSUE REQUEST SENSE
/TRANSMIT SENSE DATA

ONLY WHEN RESPONSE IS
POSSIBLE FOR FAILURE
DETECTED BY CONTROLLER

⑤ EXECUTE RETRY
(ABOUT THREE TIMES)

⑥ ISSUE ABORT COMMAND TO
I/O DEVICE

⑦ ISSUANCE OF RETRY ENDS
WITH ERROR

⑧ NOTIFY PERMANENT ERROR
TO FILE SYSTEM

⑨ ERASE DATA IN I/O BUFFER AND SAVE
ERROR CONTENTS IN SYSTEM LOG

⑩ NOTIFY ERROR VIA APPLICATION TO USER

(WHEN CONDITION ③ RELATED TO
CONTROLLER IS IMPOSSIBLE,
EXECUTE RETRY AFTER DEVICE
DRIVER MONITORS STATE FOR A
PREDETERMINED PERIOD OF TIME)

FIG.3

CLASSIFICATION OF PROCESSES ACCORDING
TO DISK SUBSYSTEM STATES
(WHEN CONTROLLER 30 OF FIG.1 IS IN FAILURE)

STATE OF FAILED CON- TROLLER	PURPOSE OF PROCESS	USAGE STATE OF SCSI BUS	SCSI-ID TRANSFER PROCESS
I/O REQUEST FROM HOST SYSTEM NOT ACCEPTED	PREVENT FAILED CONTROLLER 30 FROM ACCEPTING I/O REQUEST AND EXECUTE TRANSFER OF SCSI-ID BY NORMAL SYSTEM 40	FREE (SCSI BUS)	NORMAL CONTROLLER 40 EXECUTES INITIATOR OPERATION TO SELECT FAILED CONTROLLER 30 AND TO PREVENT FAILED CONTROLLER 30 FROM ACCEPTING I/O REQUEST. DURING THIS OPERATION, EXECUTE AND COMPLETE ADDITION OF SCSI-ID TO NORMAL SCSI PORT AND RESETTING OF FAILED SCSI PORT
		USED BY NORMAL CONTROLLER 40	SINCE FAILED CONTROLLER 30 CAN ACCEPT I/O REQUEST FROM HOST SYSTEM 10 WHILE NORMAL CONTROLLER 40 IS USING SCSI BUS, EXECUTE AND COMPLETE ADDITION OF SCSI-ID TO NORMAL SCSI PORT AND RESETTING OF FAILED SCSI PORT WHILE NORMAL CONTROLLER 40 IS EXECUTING I/O
		USED BY ANOTHER SCSI DEVICE (USAGE OF BUS OF FAILED CONTROLLER 30 CANNOT BE DISCRIMI- NATED)	EXECUTE ADDITION OF SCSI-ID TO NORMAL SCSI PORT AND RESETTING OF FAILED SCSI PORT AFTER LAPSE OF BUS MONITOR TIME [ANOTHER DEVICE] TRANSFER SCSI-ID DURING I/O OF SCSI DEVICE (SCSI BUS MAY POSSIBLY BE UNSTABLE: I/O OF SCSI DEVICE IS FINISHED IMMEDIATELY BEFORE COMPLETION OF RESETTING OF FAILED SCSI PORT AND THEN HOST ISSUES I/O PROCESS REQUEST TO FAILED CONTROLLER 30)
I/O REQUEST OF HOST SYSTEM ALREADY ACCEPTED	BEFORE ISSU- ANCE FROM DEVICE DRIVER OF BUS PESET/ RETRY TO BE EXECUTED WHEN PRESET I/O MONITOR TIME IS LAPSED AFTER I/O REQUEST, SCSI-ID IS TRANSFERRED TO NORMAL CONTROLLER 40 TO RESPOND TO RETRY	USED BY FAILED CONTROLLER 30 (USAGE OF BUS OF ANOTHER SCSI DEVICE CANNOT BE DISCRIMI- NATED)	[FAILED CONTROLLER] TRANSFER SCSI-ID DURING HANG-UP OF SCSI BUS. SCSI BUS HANG-UP IS RELEASED BY RESETTING FAILED SCSI PORT AND NORMAL CONTROLLER 40 RESPONDS TO RETRY

SCSI-ID TRANSFER PROCESS AT DETECTION OF BUS FREE STATE (CONTROLLER 30 OF FIG.1 IS IN FAILURE)

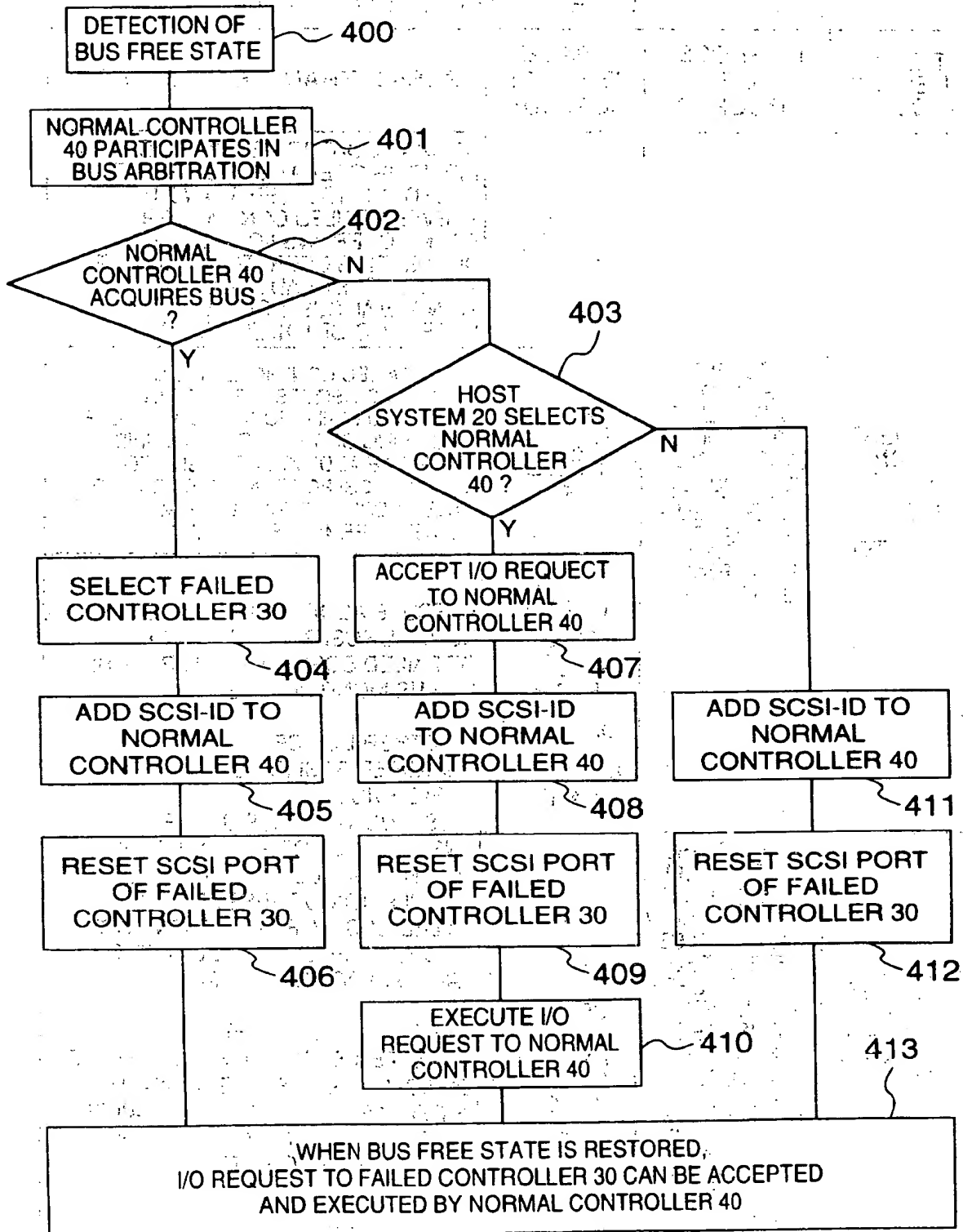


FIG.5

SCSI-ID TRANSFER PROCESS AT DETECTION OF BUS BUSY STATE
(CONTROLLER 30 OF FIG.1 IS IN FAILURE)

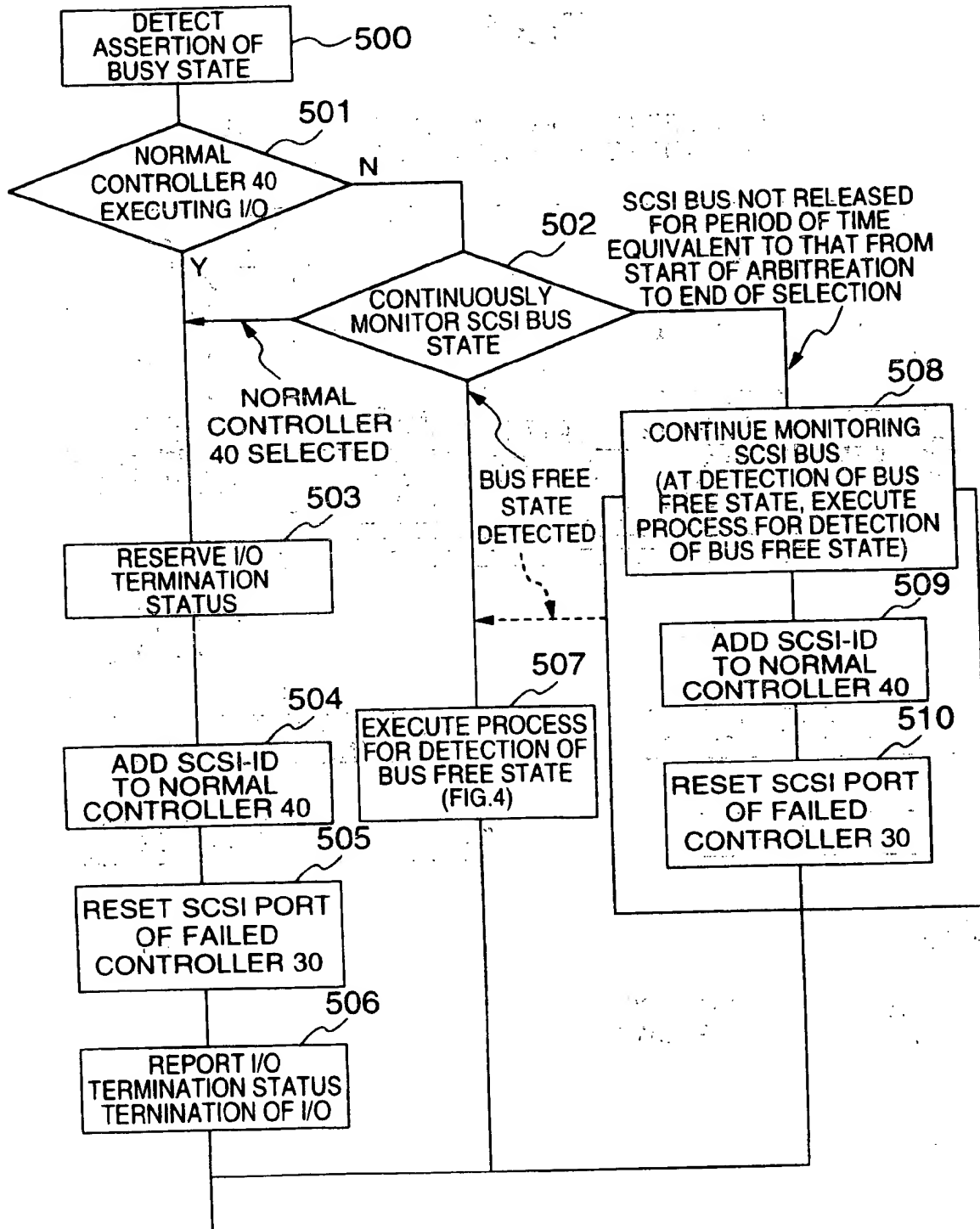


FIG.6

CONTROLLER CONFIGURATION NOT HAVING
PORT ADDRESS RESETTNG FACILITY

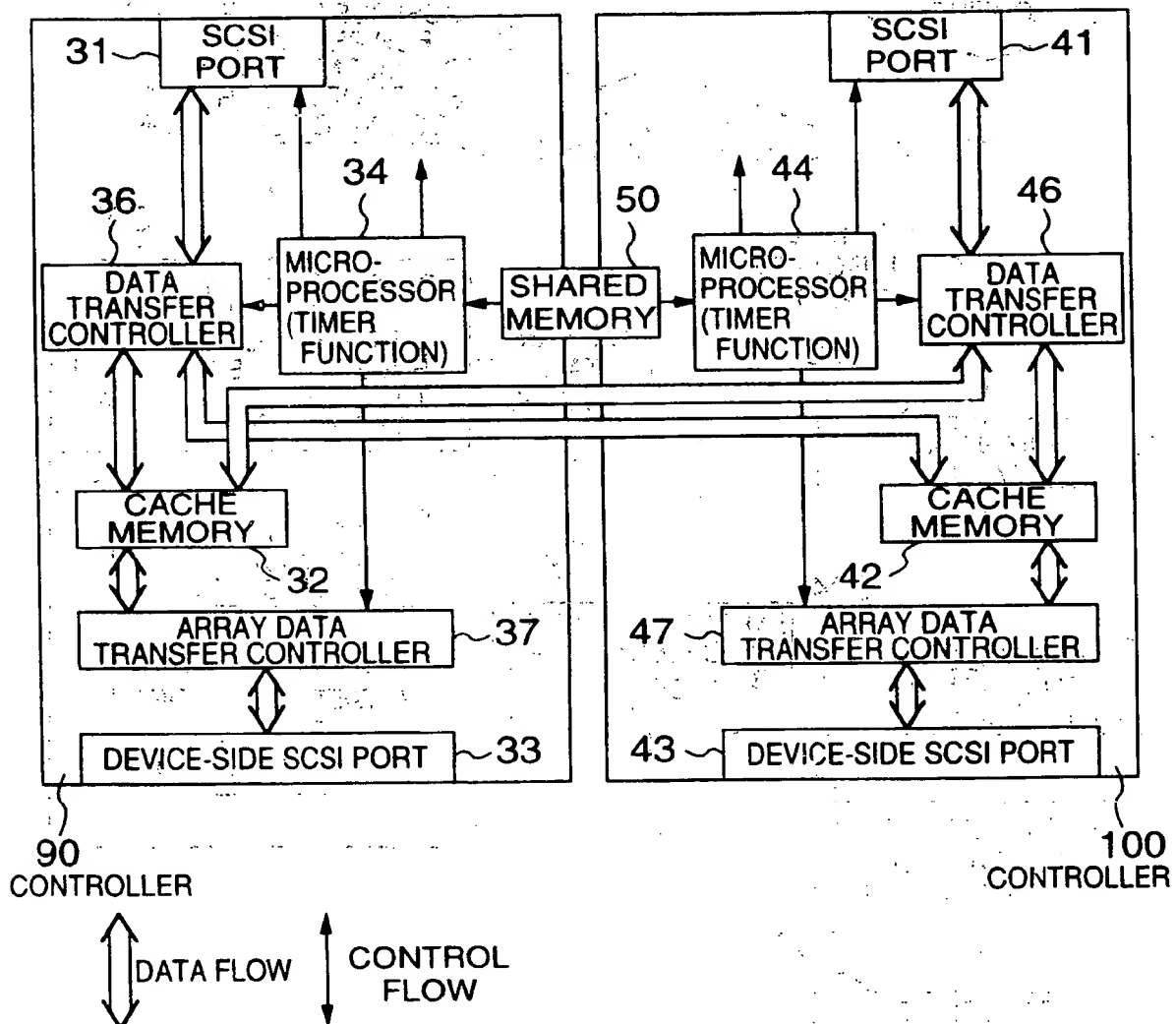
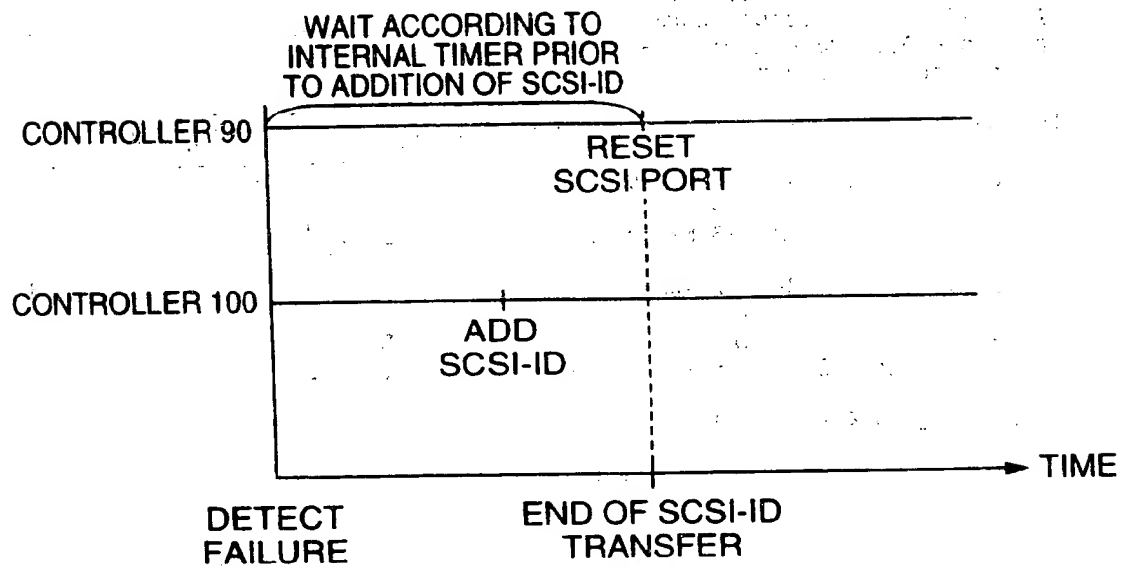


FIG.7

SCSI-ID TRANSFER IN CONFIGURATION NOT
HAVING PORT ADDRESS RESETTING FACILITY
(EXAMPLE OF I/O TRANSFER BY CONTROLLER
100 AT FAILURE IN CONTROLLER 90)





European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 96 11 7200

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
L	EP 0 747 822 A (HITACHI, LTD) 11 December 1996 Closest prior art published after filing date, cited in application * the whole document *	1-21	G06F11/20
A	WO 93 18456 A (ARRAY TECHNOLOGY CORP) 16 September 1993 * abstract; claim 1; figures 1,4 * * page 6, line 28 - page 7, line 23 * * page 10, line 28 - page 12, line 10 * * page 25, line 9 - page 28, line 2 *	1,5,9, 11,16,21	
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 16, no. 3, August 1973, NEW YORK, US, pages 912-914, XP002024376 ANONYMOUS: "Graceful Degradation in a Multiple Data Path Environment." * the whole document *	1,5,9, 11,16,21	
A	DE 38 01 547 A (HITACHI LTD) 28 July 1988 * abstract; claims 1-3,6; figure 1 *	1,5,9, 11,16,21	TECHNICAL FIELDS SEARCHED (Int.Cl.6) G06F
A	EP 0 475 624 A (IBM) 18 March 1992 * column 6, line 38 - column 7, line 16 *	1,5,9, 11,16,21	
A	PATENT ABSTRACTS OF JAPAN vol. 018, no. 039 (P-1679), 20 January 1994 & JP 05 265914 A (HITACHI LTD), 15 October 1993, * abstract *	1,5,9, 11,16,21	
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 4 February 1997	Examiner Herreman, G
<p>CATEGORY OF CITED DOCUMENTS</p> <p>X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document</p> <p>T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document</p>			

EPO FORM 1501 (03.92) (P/MCO)

THIS PAGE BLANK (USPTO)